



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/468,469	12/21/1999	REGINALD V. BLUE		3674

7590 04/28/2005

ROCCO L ADORNATO
UNISYS CORP
TOWNSHIP LINE & UNION MEETING ROADS
BLUE BELL, PA 19424

EXAMINER

ALI, SYED J

ART UNIT

PAPER NUMBER

2195

DATE MAILED: 04/28/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/468,469	BLUE, REGINALD V.
Examiner	Art Unit	
Syed J Ali	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 02 February 2005.

2a) This action is **FINAL**. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-45 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-45 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on 21 December 1999 is/are: a) accepted or b) objected to by the Examiner.

 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)
2) Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.
4) Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
5) Notice of Informal Patent Application (PTO-152)
6) Other: _____

DETAILED ACTION

1. This office action is in response to the amendment filed February 2, 2005. Claims 1-45 are presented for examination.
2. The text of those sections of Title 35, U.S. code not included in this office action can be found in a prior office action.

Claim Objections

3. **Claims 15 and 35-36 are objected to because of the following informalities:**
 - a. In line 4 of claim 15, "instruction" should read "instructions".
 - b. In line 1 of claim 35, "comprise" should read "comprises".
 - c. In line 1 of claim 36, "compute" should read "computer".

Appropriate correction is required.

Claim Rejections - 35 USC § 112

4. **Claims 11, 29, 36, and 45 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.**

5. As per claims 11, 29, 36, and 45 the claims are phrased in such a way as to present what should be independent claims as dependent claims. Any claim which is in dependent form but which is so worded that it, in fact, is not a proper dependent claim, as for example it does not

include every limitation of the claim on which it depends, will be required to be canceled as not being a proper dependent claim; and cancellation of any claim depending on such a dependent claim will be similarly required. The applicant may thereupon amend the claims to place them in proper dependent form, or may redraft them as independent claims, upon payment of any necessary additional fee. MPEP §607.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

7. **Claims 1-45 are rejected under 35 U.S.C. 102(e) as being anticipated by Oliver (USPN 6,029,190).**

8. As per claim 1, Oliver teaches the invention as claimed, including a method of managing a resource shared among concurrently executing threads in a multi-threaded computer program running under an operating system that supports multi-threaded computer programs, said method comprising the acts of:

receiving, from a first thread, a request for a lock, said request indicating whether said request is for a read lock or a write lock (col. 2 lines 52-63; col. 4 lines 15-17);

if said request is for a read lock, granting said request and permitting said thread to proceed (col. 3 lines 2-3) unless another of said threads is writing said resource (col. 2 lines 65-67); and

if said request is for a write lock, granting said request and permitting said thread to proceed (col. 3 lines 64-66) unless another of said threads is reading or writing said resource (col. 4 lines 2-4).

9. As per claim 2, Oliver teaches the invention as claimed, including the method of claim 1, wherein said request is issued by creating a local class instance, wherein a constructor for said class instance issues said request (col. 6 lines 35-38, 43-46).

10. As per claim 3, Oliver teaches the invention as claimed, including the method of claim 2, wherein said class instance is a class instance in the C++ programming language (col. 6 lines 35-38).

11. As per claim 4, Oliver teaches the invention as claimed, including the method of claim 2, further comprising the act of destroying said local class instance, wherein a destructor for said class instance issues a request to release said lock (col. 6 lines 43-46).

12. As per claim 5, Oliver teaches the invention as claimed, including the method of claim 1, further comprising determining whether other threads are reading or writing the resource,

wherein the determination of whether other threads are reading or writing from said resource are made by claiming one or more critical sections (col. 4 lines 34-36, 41-45).

13. As per claim 6, Oliver teaches the invention as claimed, including the method of claim 5, wherein said critical sections are implemented by way of a critical section facility of said operating system (col. 1 lines 25-29, see also paragraph 53 below).

14. As per claim 7, Oliver teaches the invention as claimed, including the method of claim 5, further comprising the act of incrementing a counter (col. 3 lines 6-9).

15. As per claim 8, Oliver teaches the invention as claimed, including the method of claim 7, wherein the value of said counter is the number of read locks outstanding on said resource (col. 3 lines 6-9).

16. As per claim 9, Oliver teaches the invention as claimed, including the method of claim 8, wherein the act of claiming at least one of said critical sections is conditioned upon the value of said counter (col. 3 lines 39-43, 64-66).

17. As per claim 10, Oliver teaches the invention as claimed, including the method of claim 1, wherein said resource comprises a data object located within the address space of said computer program (col. 1 lines 13-16).

18. As per claim 11, Oliver teaches the invention as claimed, including a computer-readable medium having computer-executable instructions to perform the method of claim 1 (col. 1 lines 7-10).

19. As per claim 12, Oliver teaches the invention as claimed, including a system for managing the use of a resource shared among concurrently-executing threads, said system comprising:

a record for maintaining information as to whether any of said threads is accessing a resource at a given point in time (col. 3 lines 9-16; col. 4 lines 8-10);

an object, which comprises or references:

a constructor, said constructor comprising computer-executable instructions to obtain a lock on said resource and to record said lock in said record (col. 2 lines 63-65; col. 6 lines 43-46); and

a destructor, said destructor comprising a set of computer-executable instructions to release said lock and to record the release of said lock in said record (col. 3 lines 14-16; col. 6 lines 43-46);

wherein the constructor instructions are executed upon creation of an instance of said object within a local scope, wherein the destructor instructions are executed upon the exiting of said local scope, and wherein no instruction, other than an instruction to exit said local scope, is required to release said lock (col. 6 lines 35-38, 43-46).

20. As per claim 13, Oliver teaches the invention as claimed, including the system of claim 12, wherein said constructor further comprises an instruction to claim a critical section (col. 4 lines 34-36, 41-45; col. 6 lines 35-38, 43-46), and wherein said destructor further comprises an instruction to relinquish said critical section (col. 4 lines 62-66; col. 6 lines 35-38, 43-46).

21. As per claim 14, Oliver teaches the invention as claimed, including the system of claim 13, wherein said critical section is implemented by way of a critical section facility of an operating system (col. 1 lines 25-29, see also paragraph 53 below).

22. As per claim 15, Oliver teaches the invention as claimed, including the system of claim 13, wherein said record comprises a counter, wherein said constructor further comprises an instruction to increment said counter (col. 3 lines 6-9; col. 6 lines 43-46), wherein said destructor comprises an instruction to decrement said counter (col. 3 lines 30-33; col. 6 lines 43-46), wherein said constructor further comprises instruction to condition the claiming of said critical section upon the value of said counter (col. 3 lines 35-43; col. 6 lines 43-46), and wherein said destructor further comprises instructions to condition the relinquishment of said critical section upon the value of said counter (col. 3 lines 34-37; col. 6 lines 43-46).

23. As per claim 16, Oliver teaches the invention as claimed, including the system of claim 12, wherein said object is a class object in the C++ programming language (col. 6 lines 35-38, 43-46).

24. As per claim 17, Oliver teaches the invention as claimed, including the system of claim 12, wherein said resource comprises a data object located within the address space of a computer program (col. 1 lines 13-16).

25. As per claim 18, Oliver teaches the invention as claimed, including a method of managing a resource shared among a plurality of concurrently executing threads, comprising the acts of:

claiming a first critical section, wherein said first critical section is unavailable to a thread seeking to do a write to said resource and to a thread seeking to do a read from said resource whenever any of said threads is presently writing to said resource (col. 2 lines 63-65; col. 3 lines 64-66);

if said first critical section is unavailable, waiting at least until said first critical section becomes available (col. 2 line 67 - col. 3 line 2; col. 3 line 66 - col. 4 line 4);

claiming a second critical section, wherein said first critical section is unavailable to a thread seeking to do a write to said resource whenever any of said threads is presently reading from said resource (col. 3 lines 11-14; col. 4 lines 4-10);

if said second critical section is unavailable, waiting at least until said second critical section becomes available (col. 4 lines 2-4); and

executing at least one instruction that accesses said resource (col. 3 lines 17-18; col. 4 line 11).

26. As per claim 19, Oliver teaches the invention as claimed, including the method of claim 18, wherein said threads are threads of a single multi-threaded computer program (col. 1 lines 13-16).

27. As per claim 20, Oliver teaches the invention as claimed, including the method of claim 18, wherein said critical sections are implemented by way of a critical section facility of an operating system (col. 1 lines 25-29, see also paragraph 53 below).

28. As per claim 21, Oliver teaches the invention as claimed, including the method of claim 18, wherein said at least one executed instruction that accesses said resource is a write access (col. 3 lines 44-45), and wherein said method further comprises the acts of:

relinquishing said second critical section (col. 4 lines 8-10); and
after performing said executed instruction, relinquishing said first critical section (col. 4 lines 8-10).

29. As per claim 22, Oliver teaches the invention as claimed, including the method of claim 18, wherein said at least one executed instruction that accesses said resource is a read access (col. 2 lines 50-51), and wherein said method further comprises the acts of:

relinquishing said first critical section (col. 3 lines 14-16); and
after performing said executing act, relinquishing said second critical section, unless another set of instructions is presently reading from said resource (col. 3 lines 30-37).

30. As per claim 23, Oliver teaches the invention as claimed, including the method of claim 22, wherein the determination of whether any set of instructions is presently reading from said resource is made by testing the value of a counter (col. 3 lines 39-43, 64-66).

31. As per claim 24, Oliver teaches the invention as claimed, including the method of claim 18, further comprising the acts of:

creating a local class instance (col. 6 lines 35-38, 43-46); and

after said executing said executed instruction, destroying said local class instance (col. 6 lines 35-38, 43-46);

wherein said claiming acts are invoked by the constructor for said local class instance, and wherein the destructor for said local class instance relinquishes at least one of the critical sections (col. 6 lines 35-38, 43-46).

32. As per claim 25, Oliver teaches the invention as claimed, including the method of claim 24, wherein said local class instance is a C++ class (col. 6 lines 35-38, 43-46), wherein said act of creating a local class instance comprises opening a local scope in a program in the C++ programming language (col. 6 lines 35-38, 43-46), and wherein said act of destroying said local class instance comprises closing said local scope (col. 6 lines 35-38, 43-46).

33. As per claim 26, Oliver teaches the invention as claimed, including the method of claim 18, further comprising the act of incrementing a counter (col. 3 lines 6-9), wherein said act of

claiming said second critical section is conditioned upon the value of said counter (col. 3 lines 39-43, 64-66).

34. As per claim 27, Oliver teaches the invention as claimed, including the method of claim 18, further comprising the acts of claiming and relinquishing a third critical section, wherein said third critical section is relinquished prior to executing said one instruction (col. 4 lines 62-65).

35. As per claim 28, Oliver teaches the invention as claimed, including the method of claim 18, wherein said resource comprises a data object located within the address space of a computer program (col. 1 lines 13-16).

36. As per claim 29, Oliver teaches the invention as claimed, including a computer-readable medium having computer-executable instructions to perform the method of claim 18 (col. 1 lines 7-10).

37. As per claim 30, Oliver teaches the invention as claimed, including a method of managing a resource in a computer environment that supports concurrent execution of a plurality of sets of computer-executable instructions, said method comprising:

in a one of said set of instructions:

opening a local scope (col. 6 lines 35-38, 43-46);

creating an object instance within said local scope, wherein said instance comprises or references a constructor method, and wherein said constructor method

comprises instructions to obtain a lock on said resource (col. 2 lines 63-65; col. 6 lines 35-38, 43-46);

performing, subsequent to creating said instance, one or more operations, wherein at least one of said operations reads from or writes to said resource (col. 4 lines 41-45; col. 6 lines 35-38, 43-46); and, when none of said plurality of sets of computer executed instructions seeks to read from or write to said resource,

closing said local scope, whereupon said instance is destroyed, said instance further comprising or referencing a destructor method, and wherein said destructor method comprises instructions to release said lock (col. 4 lines 62-65; col. 6 lines 35-38, 43-46).

38. As per claim 31, Oliver teaches the invention as claimed, including the method of claim 30, wherein said instructions are written in the C++ programming language, and wherein said object instance is a class instance in the C++ programming language (col. 6 lines 35-38, 43-46).

39. As per claim 32, Oliver teaches the invention as claimed, including the method of claim 30, wherein said constructor further comprises an instruction to claim a critical section (col. 4 lines 34-36, 41-45).

40. As per claim 33, Oliver teaches the invention as claimed, including the method of claim 32, wherein said sets of instructions are threads of a single multi-threaded computer program executing under an operating system (col. 1 lines 13-16), and wherein said critical sections are

implemented by way of the critical section facility of said operating system (col. 1 lines 25-29, see also paragraph 53 below).

41. As per claim 34, Oliver teaches the invention as claimed, including the method of claim 30, wherein constructor further comprises instructions to increment a counter, the value of said counter being the number of read locks outstanding on said resource (col. 3 lines 6-9).

42. As per claim 35, Oliver teaches the invention as claimed, including the method of claim 30, wherein said resource comprises a data object located within the address space of a computer program (col. 1 lines 13-16).

43. As per claim 36, Oliver teaches the invention as claimed, including a computer-readable medium having computer-executable instructions to perform the method of claim 30 (col. 1 lines 7-10).

44. As per claim 37, Oliver teaches the invention as claimed, including a method of managing a resource in a computing environment that supports concurrent execution of a plurality of sets of computer-executable instructions, said method comprising:

(a) issuing, in a first set of instructions, a first request for said first of said sets of instructions to obtain a lock on said resource, wherein said request comprises an indication as to whether said first of said sets of instructions needs a read lock on said resource or a write lock on said resource (col. 2 lines 52-63; col. 4 lines 15-17);

- (b) claiming a first critical section (col. 3 lines 2-3; col. 3 line 66 - col. 4 line 2);
(c) if said indication is that said first set of instructions needs a write lock on said resource (col. 3 lines 45-46):

(c)(1) claiming a second critical section (col. 3 line 66 - col. 4 line 2); and

(c)(2) relinquishing said second critical section (col. 4 lines 8-10);

whereupon said write lock is granted to said first set of instructions (col. 3 lines 64-66);

and

- (d) if said indication is that said first of said sets of instructions needs a read lock on said resource (col. 2 lines 50-51):

(d)(1) relinquishing said first critical section (col. 3 lines 14-16); and

(d)(2) if no other one of said plurality of sets of instructions, exclusive of said first of said sets of instructions, has a read lock on said resource, claiming said second critical section (col. 3 lines 11-12);

whereupon said read lock is granted to said first set of instructions (col. 2 lines 63-65).

45. As per claim 38, Oliver teaches the invention as claimed, including the method of claim 37, wherein said sets of instructions are threads of a single computer program executing under an operating system (col. 1 lines 13-16), and wherein said critical sections are implemented by way of the critical section facility of said operating system (col. 1 lines 25-29, see also paragraph 53 below).

46. As per claim 39, Oliver teaches the invention as claimed, including the method of claim 37, further comprising the acts of:

after said act of issuing said first request, claiming a third critical section (col. 4 lines 61-66); and

before, or contemporaneously with, the granting of a lock, relinquishing said third critical section (col. 4 lines 61-66).

47. As per claim 40, Oliver teaches the invention as claimed, including the method of claim 37, further comprising the acts of:

(e) issuing, in said first set of instructions, a second request to release said lock (col. 3 lines 21-23);

(f) if said lock is a read lock and no other one of said sets of instructions, exclusive of said first set of instructions, presently has a read lock on said resource, relinquishing said second critical section (col. 3 lines 30-37); and

(g) if said lock is a write lock, relinquishing said first critical section (col. 4 lines 12-14).

48. As per claim 41, Oliver teaches the invention as claimed, including the method of claim 40, further comprising the acts of:

creating a local class instance (col. 6 lines 35-38, 43-46); and

destroying said local class instance (col. 6 lines 35-38, 43-46);

wherein said first request is issued by the constructor for said class instance (col. 6 lines 35-38, 43-46), and said second request is issued by the destructor for said class instance (col. 6 lines 35-38, 43-46).

49. As per claim 42, Oliver teaches the invention as claimed, including the method of claim 41, wherein said class instance is a class instance in the C++ programming language (col. 6 lines 35-38).

50. As per claim 43, Oliver teaches the invention as claimed, including the method of claim 40, further comprising the acts of incrementing and decrementing a counter (col. 3 lines 6-9, 30-33), wherein the value of said counter is the number of read locks outstanding on said resource (col. 3 lines 6-9), and wherein said indication of whether any other of said sets of instructions has a read lock on said resource are made by testing the value of said counter (col. 3 lines 39-43, 64-66).

51. As per claim 44, Oliver teaches the invention as claimed, including the method of claim 37, wherein said resource comprises a data object located within the address space of a computer program (col. 1 lines 13-16).

52. As per claim 45, Oliver teaches the invention as claimed, including a computer-readable medium having computer-executable instructions to perform the method of claim 37 (col. 1 lines 7-10).

Conclusion

53. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Parry et al. (USPN 6,748,481) is cited to establish that in Win32, "critical sections" and "mutexes" are essential equivalents (col. 10 lines 1-5).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (571) 272-3769. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.


Syed Ali
April 21, 2005


MAJID BANANKHAH
PRIMARY EXAMINER